

## Physics of the Absurd

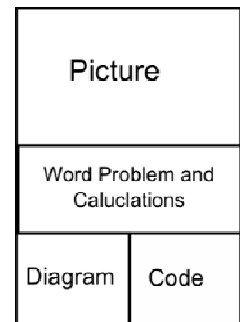
Working in groups of 2 to create weird/crazy situations related to the concept of Projectile Motion.

Project Proposal Form Requirements (due February 16<sup>th</sup>):

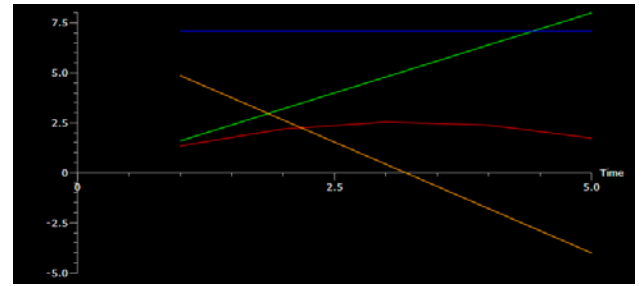
- Picture Sketch:
  - Should have at least one moving object
  - The landscape and scenery
  - The motion effect (who is going to move and how will it look)
  - Anything else, it needs to be fully designed.
- Description:
  - The word problem, explained fully and succinctly.
  - Creativity and absurdity are encouraged.
  - Spelling and grammar are correct.
- Calculations
  - A sketch of the motion that is occurring.
  - The equations involved.
  - Math is solved step by step.
  - All numbers have 3 significant figures.
  - The variables and constants.
  - A table including 5 or more data points for  $v_x$ ,  $v_y$ ,  $dx$ , and  $dy$  (depending on scenario).
- Graph:
  - On a grid.
  - $V_x$ ,  $V_y$ ,  $dx$ , and  $dy$  are all graphed as dependent on time.
  - A key clarifies which object is which.

Final Draft:

- Picture, Diagram, Description is all combined into a 11"x26" file.
- All is mounted onto a hard backed project board.
- Black border separates each section.
- Names are on the back of the board, large enough to be found.
- Layout can be found in diagram on right
- All rough draft requirements are fulfilled.
- Picture:
  - Picture is in high resolution.
  - The motion effect matches video for realism.
  - Minimum of 5 points of motion.
  - Costumes/Makeup/SFX are used
  - The background fits with the action taking place.
  - Final maximum picture size is 8"x11".



- Graph
  - VPython graph screen shot of vx, vy, dx, and dy.
  - Maximum size of diagram is 5"x5".
- Code
  - Includes comments and explanations
  - Screen shot of code and table of output
  - Utilizes looping and dt.
  - Maximum size of code is 5"x5".
- Calculations
  - Diagram matches the word problem scenario and is mathematically accurate.
  - "Microsoft Word or "Open Office" is used to produce quality formatted equations and math.
  - Font chosen is large enough and clear enough to be read from at least 3 feet away.
- Description
  - Font chosen is large enough and clear enough to be read from at least 3 feet away.
- Maximum size of description and calculations is 8"x11".



```

absurd.py - C:\Documents and Settings\pwagner\Desktop\Physics of Absurd\Phils Exam
File Edit Format Run Options Windows Help

from visual.graph import *

graph1 = gdisplay(xtitle='Time',xmax=5,xmin=0, ymax=8, ymin=-5)

xdistance = gcurve(color=color.green)
ydistance = gcurve(color=color.red)
xvelocity = gcurve(color=color.blue)
yvelocity = gcurve(color=color.orange)

t=0.226 #The total time divided by 5 time constants.
vix = 7.07 #Initial velocity in the x direction
viy = 7.07 #Initial velocity in y direction
ay=-9.8 #Acceleration in the y direction due to gravity.

print "%-4s%-6s%-6s%-6s%-6s" % ('TC','dx','dy','vx','vy')

for dt in arange(1,6): #dt allows a loop to change the time elapsed
    dx = vix*(dt*t)
    dy = (viy*(dt*t)+(0.5*ay*(dt*t)**2))
    vx = vix
    vy = viy+(ay*(dt*t))
    print "%-3d%-6.2f%-6.2f%-6.2f%-6.2f" % (dt,dx,dy,vx,vy)
    xdistance.plot(pos=(dt,dx))
    ydistance.plot(pos=(dt,dy))
    xvelocity.plot(pos=(dt,vx))
    yvelocity.plot(pos=(dt,vy))

Python Shell
File Edit Shell Debug Options Windows Help

TC dx dy vx vy
1 1.60 1.35 7.07 4.86
2 3.20 2.19 7.07 2.64
3 4.79 2.54 7.07 0.43
4 6.39 2.39 7.07 -1.79
5 7.99 1.73 7.07 -4.00
>>>

```

#### Schedule:

February 9 - Introduce Project /Choose Groups and Complete front side of Project Proposal form (due at EOC)  
 February 10 - Projectile Motion Introduction/Trig Functions (notes check at end of class)  
 February 11 - Finish Projectile Motion Introduction/Work on Calculations and Graph  
 February 12 - Work on calculations and graph. Approved Project Proposal Form by end of class.

Schedule date during the week of February 16-19 to take picture. All materials on list must be at HTHCV in order to schedule a date.

February 16 - Humanizing Haiti Exhibition  
 February 17 - Pro Mo Problems/Nerf Lab  
 February 18 - Final and formatted Calculations and Description (due at end of class)  
 February 19 - VPython Code  
 February 22 - VPython Code checked by end of class  
 February 23 - Work on Pictures  
 February 24 - Work on Pictures  
 February 25 - Free work day (finish whatever needs to be done)  
 February 26 - Project due by end of class.

All parts of the project are worth 40 points (Picture + Calculations + Graph/Diagram + Code + Description) for a total of 200 points. Extra credit is available for those who exceed the project requirements.